PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-067355
(43)Date of publication of application : 16.03.2001

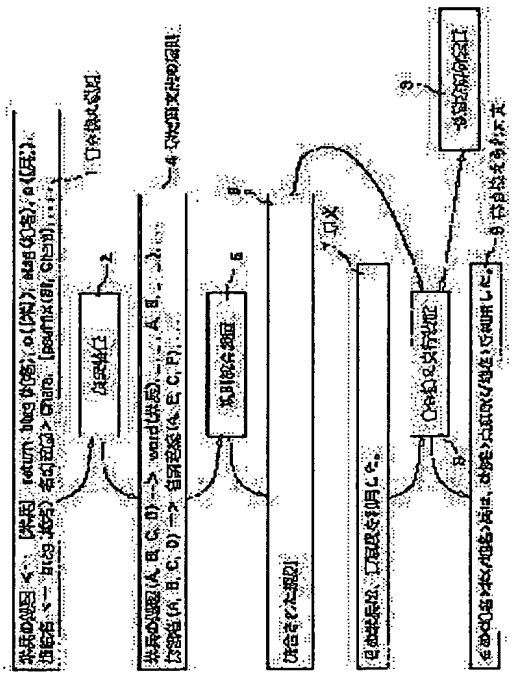
(51)Int.Cl. G06F 17/27
G06F 17/21
G06F 17/28
G06F 17/30

(21)Application number : 11-241245
(22)Date of filing : 27.08.1999

(71)Applicant : NIPPON TELEGR & TELEPH CORP <NTT>
(72)Inventor : ISOZAKI HIDEKI

(54) SENTENCE REWRITING/INFORMATION EXTRACTING METHOD WHICH ENABLES CONSTRAINED DESCRIPTION OF CHARACTER STRING, DEVICE THEREFOR AND RECORDING MEDIUM

(57)Abstract:
PROBLEM TO BE SOLVED: To briefly and declaratively specify a rewriting rule or an inherent expression extraction rule including a constraint of a character string without caring about order of processing in a system which summarizes a huge amount of document information, corrects an expression into an audible one, on the contrary, transforms a sentence written in spoken word into a written word that is easier to read, and extracts important structural elements (inherent expression) such as a person's name, a place name, a name of an organization, or time and date from a newspaper article or the like.
SOLUTION: This document rewriting transforms a set 1 of rewriting rules a user describes into a set 4 of rules of established phrase grammar by a translating device 2 and then transforms the set 4 of the rules of established phrase grammar into an integrated rule 6 which can perform high speed parallel processing. A rewriting execution device 8 receives the integrated rule 6 and a document (an original sentence 7) to be transformed and outputs a transformed result 9.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(書誌＋要約＋請求の範囲)

(19)【発行国】日本国特許庁(JP)
(12)【公報種別】公開特許公報(A)
(11)【公開番号】特開2001－67355(P2001－67355A)
(43)【公開日】平成13年3月16日(2001. 3. 16)
(54)【発明の名称】文字列制約記述可能な文章書き換え・情報抽出方法および装置ならびに記録媒体
(51)【国際特許分類第7版】
G06F 17/27 17/21 17/28 17/30
【FI】
G06F 15/20 550 A 590 E 15/38 X 15/40 370 A 15/401 320 A
【審査請求】未請求
【請求項の数】11
【出願形態】OL
【全頁数】9
(21)【出願番号】特願平11－241245
(22)【出願日】平成11年8月27日(1999. 8. 27)
(71)【出願人】
【識別番号】000004226
【氏名又は名称】日本電信電話株式会社
【住所又は居所】東京都千代田区大手町二丁目3番1号
(72)【発明者】
【氏名】磯崎 秀樹
【住所又は居所】東京都千代田区大手町二丁目3番1号 日本電信電話株式会社内
(74)【代理人】
【識別番号】100088328
【弁理士】
【氏名又は名称】金田 暢之
【テーマコード(参考)】
5B0095B0755B091
【Fターム(参考)】
5B009 QA03 QA05 QA11 QA12 QA14 QA17 VA02 5B075 ND03 NK32 NS01 NS10 UU05 5B091 AA13 AA15 BA03 BA12 CA02 DA03

(57)【要約】
【課題】膨大な文書情報を要約したり、音声で聞き取りやすい表現に直したり、逆に話し言葉で書かれた文章を読みやすい書き言葉に変換したり、新聞記事等から人名や地名や組織名や日時などの文章の重要な構成要素(固有表現)を抽出したりするシステムにおいて、文字列の制約を含む書き換え規則や固有表現抽出規則を、処理の順序を気にすることなく、宣言的かつ簡潔に指定することを可能にする。
【解決手段】翻訳装置2により、ユーザが記述した書き換え規則の集合1を確定節文法の規則の集合4に変換し、次に、規則統合装置5により、確定節文法の規則の集合4を高速並行処理の可能な統合された規則6に変換する。書き換え実行装置8は、統合された規則6と変換すべき文書(原文7)を受けとり、変換した結果9を出力する。

【特許請求の範囲】
【請求項1】 文書情報に基づき文章の書き換えおよび／または文章からの情報抽出を行う文章書き換え・情報抽出方法であって、ユーザにより宣言的に指定された、文字列の制約を含む書き換え規則および／または抽出規則を、自動的に、論理型言語で実行可能な形式の規則に変換する翻訳ステップと、前記翻訳ステップで得られた複数の規則を統合する統合ステップと、前記統合ステップで得られた統合された規則を実際の文章に適用し、前記文章の書き換えおよび／または前記文章からの情報抽出を自動的に行う適用ステップと、を有する文章書き換え・情報抽出方法。
【請求項2】 前記論理型言語で実行可能な形式の規則が、確定節文法による規則である請求項1に記載の文章書き換え・情報抽出方法。
【請求項3】 前記翻訳ステップにおいて、差分リストを用い、前記文字列の制約を含む書き換え規則および／または抽出規則を構成する非終端記号に対して、入力の文字列表現を保持するための引数と、出力の文字列表現を保持するための引数とが追加され、当該規則のボディー部に現れる構成要素の文字列表現を順に連結したものが、変換後の規則のヘッド部の非終端記号の文字列表現になるように、差分リストを構成する各変数が設定され、ボディー部に return による書き換え指定がある場合には、その return より前の部分で得られる出力を無視し、その return の後に指定された文字列を連結して得られる文字列の差分リスト表現が、変換後の規則のヘッド部の出力用の引数の差分リスト表現になるように、前記差分リストを構成する各変数が設定され、前記ボディー部に現れる終端記号が、対応する文字列を出力側にコピーするための述語の呼出しに置き換えられ、前記ボディー部に現れる付加制約が、そのまま変換後の規則に残される、請求項2に記載の文章書き換え・情報抽出方法。
【請求項4】 前記適用ステップにおいて前記文章の形態素解析を実行する請求項1乃至3いずれか1項に記載の文章書き換え・情報抽出方法。
【請求項5】 文書情報に基づき文章の書き換えおよび／または文章からの情報抽出を行う文章書き換え・情報抽出装置であって、ユーザにより宣言的に指定された、文字列の制約を含む書き換え規則および／または抽出規則を論理型言語で実行可能な形式の規則に変換する翻訳手段と、前記翻訳手段で得られた複数の規則を統合する統合手段と、前記統合手段で得られた統合された規則を実際の文章に適用し、前記文章の書き換えおよび／または前記文章からの情報抽出を行う適用手段と、を有する文章書き換え・情報抽出装置。
【請求項6】 前記論理型言語で実行可能な形式の規則が、確定節文法による規則である請求項5に記載の文章書き換え・情報抽出装置。
【請求項7】 前記翻訳手段が、差分リストを用い、前記文字列の制約を含む書き換え規則および／または抽出規則を構成する非終端記号に対して、入力の文字列表現を保持するための引数と、出力の文字列表現を保持するための引数とを追加し、当該規則のボディー部に現れる構成要素の文字列表現を順に連結したものが、変換後の規則のヘッド部の非終端記号の文字列表現になるように、差分リストを構成する各変数を設定し、ボディー部に return による書き換え指定がある場合には、その return より前の部分で得られる出力を無視し、その return の後に指定された文字列を連結して得られる文字列の差分リスト表現が、変換後の規則のヘッド部の出力用の引数の差分リスト表現になるように、前記差分リストを構成する各変数を設定し、前記ボディー部に現れる終端記号を、対応する文字列を出力側にコピーするための述語の呼出しに置き換え、前記ボディー部に現れる付加制約を、そのまま変換後の規則に残す、処理を実行するものである請求項6に記載の文章書き換え・情報抽出装置。
【請求項8】 前記適用手段と連係し前記文章の形態素解析を実行する形態素解析手段をさらに有する、請求項5乃至7のいずれか1項に記載の文章書き換え・情報抽出装置。
【請求項9】 コンピュータが読取り可能な記録媒体であって、前記コンピュータに、ユーザにより宣言的に指定された、文字列の制約を含む書き換え規則および／または抽出規則を論理型言語で実行可能な形式の規則に変換する翻訳ステップと、前記翻訳ステップで得られた複数の規則を統合する統合ステップと、前記統合ステップで得られた統合された規則を実際の文章に適用し、前記文章の書き換えおよび／または前記文章からの情報抽出を行う適用ステップと、を実行させるプログラムを格納した記録媒体。
【請求項10】 前記論理型言語で実行可能な形式の規則が、確定節文法による規則である請求項9に記載の記録媒体。
【請求項11】 前記翻訳ステップにおいて、差分リストを用い、前記文字列の制約を含む書き換え規則および／または抽出規則を構成する非終端記号に対して、入力の文字列表現を保持するための引数と、出力の文字列表現を保持するための引数とが追加され、当該規則のボディー部に現れる構成要素の文字列表現を順に連結したものが、変換後の規則のヘッド部の非終端記号の文字列表現になるように、差分リストを構成する各変数が設定され、ボディー部に return による書き換え指定がある場合には、その return より前の部分で得られる出力を無視し、その return の後に指定された文字列を連結して得られる文字列の差分リスト表現が、変換後の規則のヘッド部の出力用の引数の差分リスト表現になるように、前記差分リストを構成する各変数が設定され、前記ボディー部に現れる終端記号が、対応する文字列を出力側にコピーするための述語の呼出しに置き換えられ、前記ボディー部に現れる付加制約が、そのまま変換後の規則に残される、請求項10に記載の記録媒体。

詳細な説明

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、文書情報に基づき文章の書き換えや情報抽出を行う文章書き換え・情報抽出方法および装置に関し、特に、膨大な文書情報を要約したり、音声で聞き取りやすい表現にしたり、逆に話し言葉で書かれた文章を読みやすい書き言葉にしたり、新聞記事等から人名や地名や組織名や日時などの文章の重要な構成要素(固有表現)を抽出したりする作業を自動的に行なう必要がある場合に、文章をどう書き換えるべきか、どこからどこまでをどう抽出すべきかをユーザが簡単に指示できる方法および装置に関する。

【0002】

【従来の技術】コンピュータを利用し、自動的に、膨大な文書情報を要約したり、音声で聞き取りやすい表現にしたり、逆に話し言葉で書かれた文章を読みやすい書き言葉に変換したりする場合、単語を構成する品詞や文字種や前後の文脈を調べてどう書き換えるかをあらかじめ規則で指定し、この規則に基づいてコンピュータにより実際の文章書き換えを行う手法が一般的である。

【0003】また、文書情報、例えば新聞記事に「大島の南南東」と書いてあった場合は、「大島」が地名であることが判断できるので、「<地名>大島</地名>」のように前後に地名であることを示すタグを挿入し、「大島氏」の場合は、「大島」が人名であると判断できるので、「<人名>大島</人名>氏」のように人名であることを示すタグを挿入しておくと、その文書情報から特定の情報を抽出したり検索したりする場合に便利である。このような処理は「固有表現抽出」と呼ばれるが、文章に対するタグの挿入であるので、これも一種の文章の書き換えと見なすことができる。固有表現抽出に使用される規則を固有表現抽出規則と呼ぶ。

【0004】上述した各規則、とくに固有表現抽出規則を書き下す場合には、システムの辞書に登録されていない人名や地名などが出現しても認識できるようにしなければならない。その場合、単語がどのような文字から構成されているかが重要な手がかりになる。たとえば、辞書に「水納島」という単語が登録されていなくても、最後に「島」という文字が付いていることから、地名であろうと判断できる。

【0005】固有表現を抽出するためには、このような文字レベルの判断が重要になってくるため、とくに日本語の固有表現抽出を行なうシステムは、文字列を分解したり、逆に連結したりする機能を持つように作られている。

【0006】

【発明が解決しようとする課題】上記のような書き換え規則や抽出規則は、実際には、コンピュータプログラムの形態で記述されている。従来、これらの規則は、おもに手続き型言語で記述されて実現されており、これらの文字列レベルの処理の記述や制御の内容は、文章書き換え・情報抽出システムの開発者に委ねられている。このため、多数の規則が定められる場合には、副作用のため規則が意図した通りに作動しないことが珍しくなく、規則を記述したあとに繰り返し試験を行なう必要があった。

【0007】また、手続き型言語でなく論理型言語を用いて規則を記述することも考えられるが、文法記述にしばしば用いられる確定節文法を用いて論理型言語により同様の規則を宣言的に記述しようとしても、単語を文字列に分割したり、逆に文字列を結合したり、入力側の各単語を出力側にコピーしたりといった手続き的処理を文法中に記述しなければならず、各規則が繁雑で分かり難く、誤りが混入しやすい、といった問題があった。

【0008】本発明は、上記に鑑みてなされたもので、その目的とするところは、文字列レベルの判断を含む処理を、文字列に関する制約として宣言的かつ簡潔に記述できる処理方法および装置を提供することにある。

【0009】

【課題を解決するための手段】上記目的を達成するため、本発明の文章書き換え・情報抽出方法では、システムの目的(どのような書き換えを行いたいか、どのような情報を抽出したいかなど)に応じて、あらかじめ書き換え規則あるいは抽出規則の集合をユーザが宣言的に与えておく。その上で、自動的に、これらの書き換え規則、抽出規則を論理型言語で実行可能な形式に変換し、変換後の複数の規則を統合して高速処理可能な形式に統合する。実際に文書情報の書き換えや文書情報からの情報抽出等を行う際には、統合された規則を実際の文章に適用する。ここで、論理型言語で実行可能な形式の規則は、典型的には、確定節文法による規則である。

【0010】ユーザが与えた書き換え規則、抽出規則を論理型言語で実行可能な形式に変換するステップすなわち翻訳ステップの具体例を説明する。翻訳ステップは、各規則を構成する非終端記号に対して、入力の文字列表現を保持するための引数と、出力の文字列表現を保持するための引数とを追加する。これらの引数は、それぞれ、いわゆる差分リストを用いることにより実装することができる。したがって、各非終端記号には、4つの引数が追加されることになる。そして、規則のボディー部に現れる構成要素の文字列表現を順に連結したものが、ヘッド部の非終端記号の文字列表現になるように、差分リストを構成する各変数を設定する。

【0011】ここで、ボディー部に return による書き換え指定がある場合には、その return より前の部分で得られる出力を無視し、return の後に指定された文字列を連結して得られる文字列の差分リスト表現が、ヘッド部の出力用の引数の差分リスト表現になるように、差分リストを構成する各変数を設定する。

【0012】なお、ボディー部に現れる終端記号は、対応する文字列を出力側にコピーするための述語の呼出しに置き換える。また、ボディー部に現れる付加制約は、そのまま翻訳後の規則に残す。

【0013】以上が、翻訳ステップの具体的な内容である。

【0014】本発明の文章書き換え・情報抽出装置は、ユーザが与えた書き換え規則や抽出規則を論理型言語で実行可能な形式に変換するための翻訳手段と、複数の規則を統合して高速処理の可能な形式に変換するための統合手段と、さらに統合された規則を実際の文章に適用するための適用手段とを有する。

【0015】(作用)本発明によれば、ユーザが書き換え規則あるいは抽出規則の集合をシステムに与えると、システムがこれを論理型言語で実行可能な形式に変換し、これら規則を統合してさらに高速処理の可能な形式に変換し、統合後の規則を用いて実際に文章の書き換え処理あるいは情報抽出処理を行なうので、ユーザは、文字列レベルの判断を含む処理を、文字列に関する制約として宣言的かつ簡潔に記述できるようになる。

【0016】

【発明の実施の形態】次に、本発明の好ましい実施の態様について、図面を参照して説明する。

【0017】図1は、本発明の実施の一形態における文章書き換え・情報抽出装置の構成を示すブロック図であり、図2は、図1に示す装置を使用し、本発明に基づく文章書き換え・情報抽出方法を実施する際の処理手順を示すデータフロー図である。

【0018】図1に示す装置は、ユーザが与えた書き換え規則や抽出規則を論理型言語で実行可能な形式に変換するための翻訳手段である翻訳装置2と、複数の規則を統合して高速処理の可能な形式に変換するための統合手段である規則統合装置5と、統合された規則を実際の文章に適用するための適用手段である書き換え実行装置8と、書き換え実行装置8により文章の書き換えを行う際に文章の形態素解析を行う形態素解析装置3と、各規則を格納する規則格納部11と、書き換え実行装置8に与える文章を格納する入力文書格納部12と、書き換えられ文章を格納する出力文書格納部13と、を備えている。規則格納部11は、具体的には、ユーザが指定した書き換え規則(または抽出規則)の集合1(図2)と、翻訳装置2で変換された後の確定節文法の規則の集合4(図2)と、規則統合装置5によって統合された後の規則6(図2)とを格納する。

【0019】次に、この文章書き換え・情報抽出装置を用いた文章書き換え・情報抽出について、図2のデータフロー図を用いて説明する。

【0020】まず、翻訳装置2が、ユーザが指定し規則格納部11に格納されている書き換え規則(あるいは抽出規則)集合1を確定節文法での規則の集合4に変換する。さらに確定節文法での規則の集合4は、規則統合装置5によって高速処理の可能な形式(統合された規則6)に変換される。統合された規則6は、書き換え実行装置8が、入力文書格納部12内の原文7を書き換えるのに利用され、書き換えた結果9は、出力されて出力文書格納部13内に格納される。

【0021】なお、ユーザの与える規則は、論理型言語で文法解析を行なう場合に一般的に用いられている「確定節文法(DCG;Definite Clause Grammer)」に準じた記法を用いるものとする。ただし、通常の確定節文法が入力に関する記述しか扱わないのに対して、本実施の形態における規則は、出力に関する記述も扱うことができる。

【0022】まず、名詞がひとつ以上連続して出現した場合にマッチする「名詞連続」という概念は、確定節文法を用いた場合と同様に、以下のように定義できる。

【0023】

【数1】名詞連続 <-- 名詞,(名詞連続;[]).これは、「名詞」の直後に「名詞連続」かまたはなにもなければ、それ全体を名詞連続と見なせることを表わす再帰的な定義になっている。この規則は出力について何も記述していないが、ここで用いる規則では、出力について記述のない規則は、入力がそのまま出力にコピーされるものとする。したがって、この定義にマッチした部分があれば、それがそのまま出力へコピーされる。

【0024】次に、固有表現抽出の例として、この名詞連続に対応する入力を文字列として見た場合に、その末尾に「駅」や「会館」などの施設を示す文字列であって、「特急通過駅」などの普通名詞的なものでなければ、全体を「地名」というタグでくくる、という固有表現抽出規則は、この実施の形態によれば、たとえば以下のように書くことができる。

【0025】

【数2】

施設名 <-- btag(地名),名詞連続>Chars, {psuffix(Sf,Chars),施設のサフイックス(Sf), not 普通名詞的な施設(Chars)},etag(地名).まず、最初の「btag(地名)」と最後の「etag(地名)」は、入力がこの規則とマッチした場合に、出

力のこの位置に、それぞれ「<地名>」、「</地名>」というタグが挿入されることを示す(btag(x)は、xについての開始タグ「<x>」を挿入することを表わし、etag(x)は、xについての終了タグ「</x>」を挿入することを表わしている)。「名詞連続>Chars」は、上記で定義した「名詞連続」とマッチした部分の入力の文字列を、この規則では Chars と呼ぶことを意味している。次の「{...}」の部分は、この文字列 Chars に関する制約を論理型言語で実行可能な形式で表わしている。具体的には、文字列 Chars のうしろの方(サフィックス)に、「駅」などの施設を表わす文字列 Sf が付いていて、しかも Chars が「特急通過駅」のような普通名詞的なものではない、という制約を表わしている。つまり、この規則では、「名詞連続」によって入力が出力にコピーされるとともに、その前後に btag, etag が出力にタグを挿入している。{...}の部分は制約を記述しているだけであり、出力には影響しない。

【0026】なお、ここで用いられた制約を記述するために用いられた記号類は、実装に用いる論理型言語を使って、次のように直接定義することができる。

【0027】施設のサフィックス([駅]).施設のサフィックス([会,館]).普通名詞的な施設([特,急,通,過,駅]).これらの制約は単語列ではなく、文字列に関するものであるため、書き換え実行装置8が形態素解析装置3を呼出して原文7の形態素解析を行わせた際の、原文7をどのように文章を分割するかに依存しにくい、という特徴を有する。ここでたとえば、これらの制約をたとえば次のように単語列で書き、上記の規則もこれに合わせて書き換えたとしよう。

【0028】普通名詞的な施設([特急,通過,駅]).

すると、形態素解析装置3が原文7を「特急」と「通過」と「駅」に分割した場合はよいが、もし、「通過駅」という単語が登録されていて、「特急」と「通過駅」に分割されてしまうと、この制約とはマッチしないため、意図しない結果が得られてしまうことになる。このように、本実施の形態では、単語列ではなく、制約を文字列で記述することによって、予期しない単語分割によって規則が適用されないという問題をかなり回避することができる。

【0029】さて、書き換え規則によっては、入力がそのまま出力にコピーされることが望ましくないことがある。たとえば、入力に「米兵」という単語がある場合、この単語を分割して、その「米」だけに「地名」というタグを付けたい場合がある。この場合はどう出力したいかを明確にするため、以下のように return のあとに出力を明記する。

【0030】

【数3】米兵の処理 <-- [米兵] return btag(地名),o([米]),etag(地名),o([兵]).最初の「[米兵]」は、「米兵」という単語が入力に現れた場合という制約を表わしている。規則中に return があると、システムはその左側に書かれた入力を出力にコピーすることはせず、return のあとに書かれた指示に従い出力する。ここでは o(...)によって、「米」という文字と「兵」という文字を別々に出力することを指示している。これによって、「<地名>米</地名>兵」という出力が得られる。

【0031】もし、「米兵」を「<地名>アメリカ</地名>兵」と書き換えたいならば、以下のように書けばよく、本発明の方法が固有表現抽出だけでなく、一般の書き換えにも使えることが分かる。

【0032】

【数4】

米兵の処理 <-- [米兵]return btag(地名),o([アメリカ]), etag(地名),o([兵]).翻訳装置2は、以上のような形式によりユーザによって書かれた規則を読み込んで、論理型言語で実行可能な確定節文法の規則に変換する。上記の各規則は以下のような規則に変換される。

【0033】

【数5】

名詞連続(A,B,C,D) --> 名詞(A,E,C,F), {名詞連続(E,G,F,H),{x(B,D)=x(G,H)}; {x(B,D)=x(E,F)}}). 施設名(A,B,C,D) --> 名詞連続(A,B,E,F),{diff(A,B,G)}, {psuffix(H,G),施設のサフィックス(H),＼+普通名詞である施設(G)}, {c=['<地名>'|E],F=[</地名>'|D]}. 米兵の処理(A,B,C,D) --> word(米兵,_,_,A,B,_,_),{c=['<地名>'|E],append([米],F,E), F=['</地名>'|G],append([兵],D,G)}).これらの確定節文法による規則をその入力であるユーザが与えた書き換え規則と比べると、ユーザが与える規則の方が、変数やリスト処理が隠されているため、単純かつ明確に規則を記述できていることが分かる。すなわち、本実施の形態においては、ユーザに対して上述したようにして規則の記述を許すことにより、ユーザは、単純かつ明確に規則を記述できることになる。

【0034】変換後の規則に現れる「施設名」や「名詞連続」などのいわゆる「非終端記号」は、それぞれ4つ、あるいはそれ以上の引数を持つ。これらの引数のうち最後の2つの引数(たとえば施設名や名詞連続の C, D)は、その部分の出力を表わすいわゆる「差分リスト」である。つまり、C が「[<地名>,東京,駅,</地名>,で,待つ,。]」で、D が「[で,待つ,。]」というリストであれば、C,D はその差の「[東京,駅]」というリストを表わしている。さらにその前の2つの引数(たとえば A,B)は、その差によって、その部分に対応する入力の文字列リスト「[東,京,駅]」を表わしている。「diff(A,B,G)」は、たとえば A が「[東,京,駅,で,待,つ,。]」であり、B が「[で,待,つ,。]」であるときに、その差の「[東,京,駅]」を G を代入する。この文字列リストが文字列制約の処理を行なうときに利用される。

【0035】「word(W,F1,F2,A,B,C,D)」は、W という単語がその場所に出現したことを表わす。F1 や F2 は形態素解析装置3が出力した単語Wの特徴であり、この例では、F1 が漢字やカタカナといった文字の種類、F2 が品詞や活用形などの情報を記録してあると仮定している。したがって、本実施形態での規則に word(_,カタカナ,_)と書いておけば、任意のカタカナ単語とマッチする。また、word(_,_,接尾辞)と書いておけば。任意の接尾辞とマッチする。逆に[米兵]は word(米兵,_,_)の略記と見なせる。確定節文法の規則に変換されたときに加わる word の最後の4つの引数は、非終端記号の場合と同じく、入力の文字列表現の差分リストと出力の差分リストである。

【0036】なお、word は、実際に、入力文字列を出力へ自動的にコピーする作業を行なう機能を有する。つまり、word(T,F1,F2,A,B,C,D) は、単語 T の文字列表現を入力側から出力側にコピーするため、diff(A,B,L),append(L,D,C)を実行する。たとえば、T が「東京駅」であれば、入力の文字列表現を示す A は「[東,京,駅|B]」である。word の diff(A,B,L),append(L,D,C)の処理によって、C が「[東,京,駅|D]」になる。この自動的なコピー機能がはじめから組み込まれているため、本実施の形態の方法では、出力に関する記述が簡単である。

【0037】さて、形態素解析装置3が出力した単語と品詞情報のリストを受けとる引数は、これらの確定節文法上には明示されていないが、通常の確定節文法に従い、引数の最後に差分リストの形で加わる。つまり、たとえば「施設名(A,B,C,D)」は論理型言語で処理する時に「施設名(A,B,C,D,E,F)」のように、形態素解析の結果を受け取るための引数 E,F が加えられる。

【0038】以上のようにして得られた確定節文法の規則4は、上昇型チャート法などの既存技法を用いることにより、すべての可能性を効率よく並行して計算することができる。そこで規則統合装置5が、確定節文法の規則4を効率よく並行計算できる規則群に変換し、書き換え実行装置8がこれを実際に文書に適用して書き換えを行なう。

【0039】なお、複数の書き換え方がある場合は、書き換え実行装置8がその中から最適なものを一つ選び出して適用する。最適なものの判断基準としては、たとえば入力の先頭から何文字あるいは何単語とその規則がマッチするかを比べて、そのうちでもっとも長くマッチするものを選ぶ、いわゆる最長一致法を用いることができる。

【0040】翻訳装置の処理フロー次に、翻訳装置2での具体的な処理フローを説明する。ユーザにより入力されこの翻訳装置2が扱う規則は、以下のいずれかの形である。

【0041】

【数6】

Head <-- InputSeq return outputSeq.Head <-- InputSeq.ここで、Head は一つの原子式である。また Inputseq は、1)終端記号、2)非終端記号、3)終端記号または非終端記号の列に文字リスト化演算子の付いたもの、4)実装に用いた論理型言語の直接呼び出し、5)出力コマンド、の任意の列である。文字リスト化演算子はたとえば G>L のような形をしたもので、G が(非)終端記号、L が文字リストである。OutputSeq は出力コマンドだけの列である。

【0042】上記の実施例では、A または B であることを示す(A;B)という表記が用いられているが、処理の説明が必要以上に複雑になるため、ここでは考えない。このような規則は A の場合と B の場合の2つの規則に分けて記述することにより、この処理フローで扱える形式になる。

【0043】翻訳装置2は、この形式の規則で書かれた書き換え規則1を読み込み、各規則を以下の要領で確定節文法の規則4に翻訳する。

【0044】1. 書き換え規則をひとつ読む。

【0045】2. 新しい論理変数 A,B,C,D を用意して、その Head にこれらを引数として追加し、出力する確定節文法の規則の頭部とする。A,B は入力の文字列表現の差分リストを表わすのに使われ、C,D は出力用の差分リストを表わすのに使われる。

【0046】3. 入力を処理する部分をまとめるための場所 INPUTPART と、出力を処理する部分をまとめる場所 OUTPUTPART を用意し、それぞれ空にしておく。

【0047】4. 規則が return を含まない規則の場合は、後述する手続き trans(A,C,InputSeq)を呼び、最終的に得られた INPUTPART と OUTPUTPART の値をこの順序で連結して出力する規則の本体とする。そして手続き trans の処理の最後の VI の値を B、最後の VO の値を D と単一化する。

【0048】5. return を含む規則の場合は、以下のように入力側の処理と出力側の処理を分けて行なう。

【0049】5－1. まず、入力側の処理のため、手続き trans(A,_,Inputseq)を呼び、手続き trans の処理の最後の VI の値を B と一致させる。ここで、_はいわゆる匿名変数である。つまり、新しい論理変数であるが、その値を参照する必要がないので、名前を付けていない。これは、InputSeq を出力に使わないからである。

【0050】5－2. 次に、出力側の処理のため、outputseq に対して手続き trans(_,C,OutputSeq)を呼び、手続き trans の処理の最後の VO の値を D と単一化する。OutputSeq は入力に使わないので、やはり入力側に匿名変数を利用している。

【0051】以上の処理により、最終的に得られた INPUTPART と OUTPUTPART の値をこの順序で連結して、出力

する規則の本体とする。そして手続き trans(A,_,InputSeq)の処理の最後の VI の値を B、手続き trans(_,C,OutputSeq)の最後の VO の値を D と単一化する。

【0052】手続き trans(A,C,Seq)次に、手続き trans(A,C,Seq)について説明する。手続き trans(A,C,Seq)では、引数の受渡しを行うため、入力の現在の先頭を表わす論理変数名を記録するためのローカル変数 VI と、出力の現在の先頭を表わす論理変数名を記録するためのローカル変数 VO を準備する。VI の初期値を A、VO の初期値を C とする。あとは、以下の処理1～5を Seq が空になるまで繰り返す。

【0053】1. seq.の先頭の要素 E が終端記号[T]であるとき、T をそのまま出力側にコピーするため、新しい論理変数 NI と NO を用意して、word(T,_,_,VI,NI,VO,NO)という記号を INPUTPART の末尾に連結する。NI,NO は、T と次の単語の境界の場所を指す論理変数として使われる。そして、VI,VO の新しい値として NI,NO を採用し、E を InputSeq から削除する。

【0054】2. Seq の先頭の要素Eが非終端記号 NT であるとき、新しい論理変数 NI と NO を用意して、NT に引数 VI,NI,VO,NI を加えたものを INPUTPART の末尾に連結する。NI,NO は、NT と次の単語の境界の場所を指す論理変数として使われる。そして、VI,VO の新しい値として NI,NO を採用し、E を InputSeq から削除する。

【0055】3. Seq の先頭の要素 E が文字リスト化演算子の付いたものG＞Lであるとき、まず手続き trans(VI,VO,G)を呼ぶ。さらに手続き trans を呼び出す時点での VI の値を VIa、trans が終了した時点での VI の値を VIb として、{diff(VIa,VIb,L)}を INPUTPART の末尾に連結する。

【0056】4. Seq の先頭の要素Eが実装に用いた論理型言語の直接呼び出し(G)である場合、G をそのまま INPUTPART の末尾に連結する。これはこれまでの入力に関する制約の追加にすぎないので、VI,VO の値は変化させない。

【0057】5. Seq の先頭の要素 E が出力コマンド o(X)である場合、新しい論理変数 NO を用意して、OUTPUTPART の末尾に{append(x,NO,VO)}を追加する。同様に、btag(X)の場合は{VO ={'<X>'|NO]}を、etag(X)の場合は{VO ={'</X>'|NO]}を OUTPUTPART の末尾に追加する。そして VO の新しい値として NO を採用する。これらは出力だけに関するもので、入力には関係ないので、VI の値は変化させない。

【0058】以上により、手続き trans(A,C,Seq)の処理が終了する。

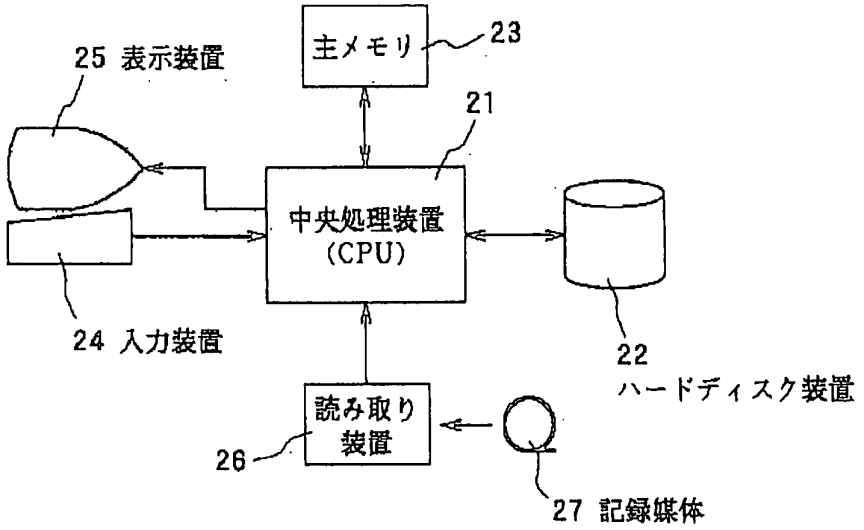
【0059】以上、本発明の好ましい実施の形態について説明したが、上述した文章書き換え・情報抽出装置は、それを実現するための計算機プログラムを、スーパーコンピュータやエンジニアリングワークステーション(EWS)などの計算機に読み込ませ、そのプログラムを実行させることによっても実現できる。文章書き換え・情報抽出を行うためのプログラムは、磁気テープやCD－ROMなどの記録媒体によって、計算機に読み込まれる。図3は、上述の文章書き換え・情報抽出方法を実行する計算機の構成を示すブロック図である。

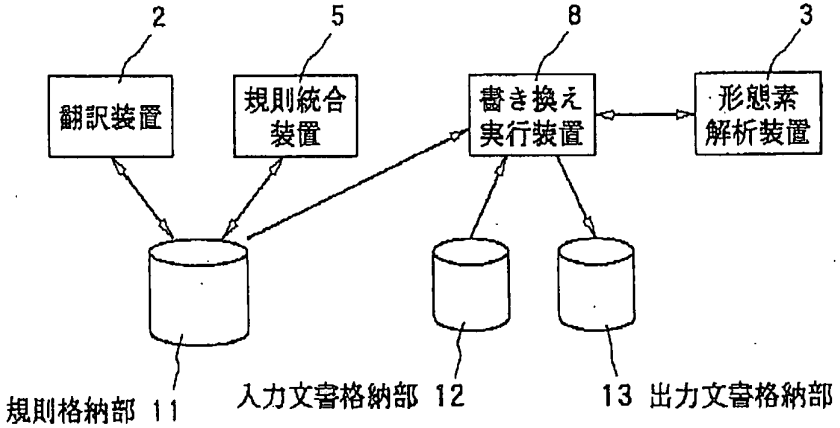
【0060】この計算機は、中央処理装置(CPU)21と、プログラムやデータを格納するためのハードディスク装置22と、主メモリ23と、キーボードやマウスなどの入力装置24と、CRTなどの表示装置25と、磁気テープやCD－ROM等の記録媒体27を読み取る読み取り装置26とから構成されている。ハードディスク装置22、主メモリ23、入力装置24、表示装置25及び読み取り装置26は、いずれも中央処理装置21に接続している。この計算機では、文章書き換え・情報抽出を行うためのプログラムを格納した記録媒体27を読み取り装置26に装着し、記録媒体27からプログラムを読み出してハードディスク装置22に格納し、ハードディスク装置22に格納されたプログラムを中央処理装置21が実行することにより、文章書き換え・情報抽出が実行される。

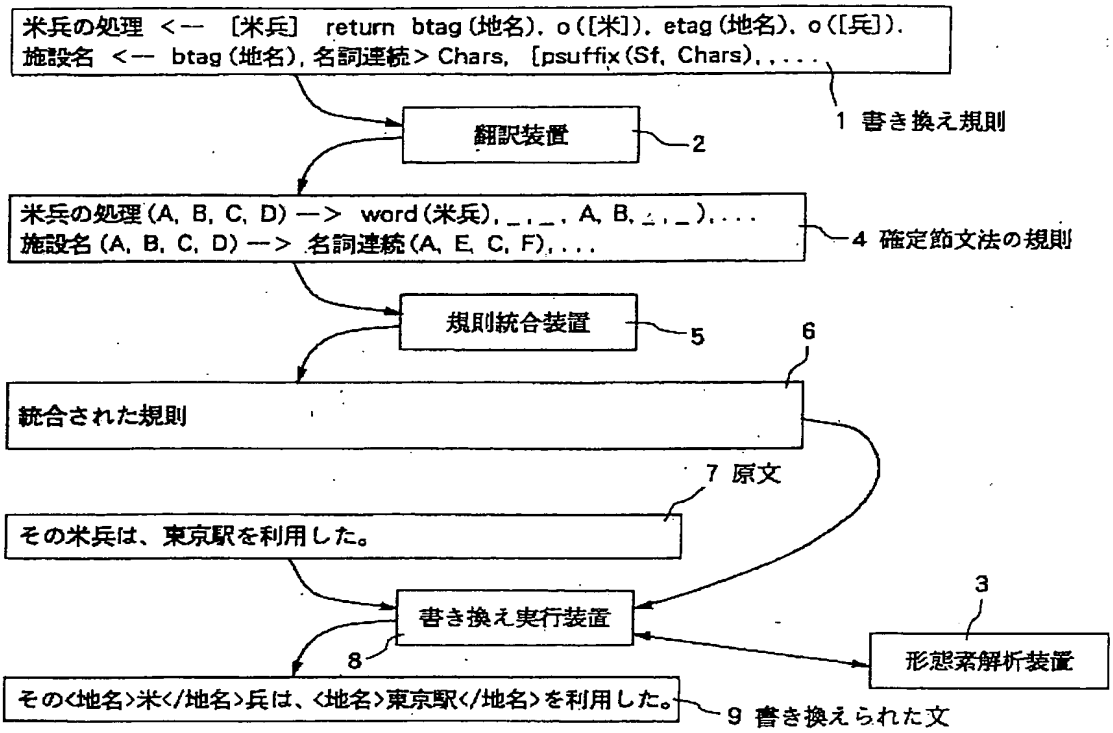
【0061】図1に示した文章書き換え・情報抽出装置との対応関係を説明すれば、翻訳装置2、形態素解析装置3、規則統合装置5および書き換え実行装置8は、中央処理装置21におけるプログラムの実行により実現でき、また、規則格納部11、入力文書格納部12および出力文書格納部13は、ハードディスク装置22や主メモリ23上に構築される。

【0062】

【発明の効果】以上説明したように、本発明は、膨大な文書情報を要約したり、音声で聞き取りやすい表現にしたり、逆に話し言葉で書かれた文章を読みやすい書き言葉にしたり、新聞記事等から人名や地名や組織名や日時などの文章の重要な構成要素(固有表現)を抽出したりするシステムにおいて、文字列の制約を含む書き換え規則や抽出規則を、処理の順序を気にすることなく、宣言的かつ簡潔に指定することが可能になるという効果がある。






米兵の処理 <— [米兵] return btag(地名), o([米]), etag(地名), o([兵]).
施設名 <— btag(地名), 名詞連続> Chars, [psuffix(Sf, Chars), ...
1 書き換え規則
翻訳装置
2
米兵の処理(A, B, C, D) —> word(米兵), _, _, A, B, _, _), . . .
施設名(A, B, C, D) —> 名詞連続(A, E, C, F), . . .
4 確定節文法の規則
規則統合装置
5
6
統合された規則
7 原文
その米兵は、東京駅を利用した。
書き換え実行装置
3
形態素解析装置
8
その<地名>米</地名>兵は、<地名>東京駅</地名>を利用した。
9 書き換えられた文